WE CLAIM:

1. A programmable logic device, comprising:

a configuration memory operable to store configuration data;

a checksum calculation engine operable to cyclically process the configuration data during operation of the programmable logic device using an error detection algorithm, the checksum calculation engine calculating a checksum during each calculation cycle, and

a checksum comparator configured to compare the checksum calculated by the checksum calculation engine in a given calculation cycle with a previously-calculated checksum so as to verify the integrity of the configuration data.

2. The programmable logic device of claim 1, wherein the checksum calculation engine is a CRC calculation engine, and wherein the checksum comparator is a CRC checksum comparator.

3. The programmable logic device of claim 1, wherein the checksum calculation engine is a parity bit calculation engine, and the checksum comparator is a parity bit checksum comparator.

4. The programmable logic device of claim 1, wherein the configuration memory is a volatile configuration memory operable to store a first type of configuration data that may be reconfigured during operation of the programmable logic device and a second type of configuration data that will not be reconfigured

during operation of the programmable logic device, the programmable logic device further comprising a self-verification control module for controlling the checksum calculation engine to exclude from processing the first type of configuration data.

5. The programmable logic device of claim 2, wherein the CRC calculation engine comprises a linear feedback shift register (LFSR).

6. The programmable logic device of claim 2, further comprising:

a register operable to store a predetermined CRC checksum, wherein the previously-calculated checksum used by the CRC checksum comparator is the predetermined CRC checksum.

7. The programmable logic device of claim 1, wherein the checksum calculation engine is configured to calculate an initial checksum during its initial cycle, and wherein the previously-calculated checksum used by the checksum comparator is the initial checksum.

8. The programmable logic device of claim 4, wherein the configuration data of the first type are associated with test bits, and wherein the self-verify control module is responsive to the test bits to control the checksum calculation engine to exclude from processing the first type of configuration data.

9. The programmable logic device of claim 1, further comprising:

a configurable logic core, wherein the checksum calculation engine and the checksum comparator are each implemented by configuring the logic core.

10. The programmable logic device of claim 1, wherein the checksum calculation engine and the checksum comparator each comprises dedicated hardware.

11. A programmable logic device, comprising:

a configuration memory operable to store configuration data; and

error-checking means for cyclically processing the configuration data during operation of the programmable logic device with an error detection algorithm so as to calculate a checksum during each calculation cycle and for comparing the checksum calculated in a given calculation cycle with a previously-calculated checksum.

12. The programmable logic device of claim 11, wherein the error-checking means is configured to cyclically process the configuration data during operation of the programmable logic device with a CRC algorithm so as to calculate a CRC checksum during each cycle and to compare the CRC checksum calculated in a given cycle with a previously-calculated CRC checksum.

13. The programmable logic device of claim 12, wherein error-checking means is configured to compare the CRC checksum calculated in a given cycle with a predetermined CRC checksum.

14. The programmable logic device of claim 12, wherein the configuration memory is a volatile configuration memory operable to store a first type of

configuration data that may be reconfigured during operation of the programmable logic device and a second type of configuration data that will not be reconfigured during operation of the programmable logic device, and wherein the error-checking means is configured to cyclically process only the second type of configuration data.

15. A method, comprising:

(a) configuring a programmable logic device with configuration data; operating the configured programmable logic device;

(b) during operation of the programmable logic device, cyclically processing the configuration data using an error-detection algorithm to generate a checksum during each calculation cycle; and

(c) after each calculation cycle, comparing the generated checksum with a previously-calculated checksum to verify the integrity of the configuration data.

16. The method of claim 1, wherein act (b) comprises cyclically processing the configuration data using a CRC algorithm to generate a CRC checksum during each cycle, and wherein act (c) comprises comparing the generated CRC checksum with a previously-calculated CRC checksum.

17. The method of claim 15, wherein act (a) comprises configuring the programmable logic device with configuration data such that the programmable logic device is configured to perform acts (b) and (c).

18. The method of claim 15, wherein the configuration data comprises a first type of configuration data that may be reconfigured during operation of the programmable logic device and a second type of configuration data that will not be reconfigured during operation of the programmable logic device, and wherein act (b) comprises cyclically processing only the second type of configuration data.

19. The method of claim 15, further comprising:

asserting a good configuration memory flag if the comparison in act (c) indicates that the integrity of the configuration memory has not been corrupted.

20. The method of claim 19, further comprising:

if the good configuration memory flag is not asserted, reconfiguring the programmable logic device with the configuration data.

21. A method of verifying the integrity of configuration data within a programmable logic device (PLD) during operation of the device, the method comprising:

providing in the device a first value derived from the configuration data at the time the configuration data is written into the PLD

calculating within the PLD a second value derived from the configuration data; and

comparing within the PLD the first value to the second value.

22. The method of claim 21, wherein the first and second values are derived according to a same error detection algorithm.

23. The method of claim 21, wherein the first and second values are checksums.

24. A programmable logic device, comprising:

a first value stored within the PLD and derived from the configuration data at the time the configuration data is written into the PLD;

a calculating circuit within the PLD and operable to calculate during operation of the PLD a second value derived from the configuration data; and

a comparator within the PLD and operable to compare during operation of the PLD the first value to the second value.